

AMENDMENTS TO THE CLAIMS

1. (Canceled)

2. (Currently Amended) A method ~~of automatically generating a consistent user interface for an application program, the method~~ comprising the computer-implemented steps of: automatically generating a consistent user interface for an application program by assisting a user with building an HTML user interface page by:

- receiving one or more business objects that each define a user action for the application program;
- receiving one or more metadata elements for dynamic content generation and defining parameters for the user actions of the one or more business object;
- invoking a controller that is communicatively coupled to one or more actions, one or more widgets, and one or more panels;
- receiving a first user request from the user through a browser used to interact with the application program and dispatching the user request to one or more of the actions;
- based on a second user request received from the user through the browser, selecting a first panel from the one or more panels and including the first panel in the HTML user interface page;
- wherein the first panel is dynamically generated using the one or more widgets and comprises a programmed template with a user-interface content;
- wherein the user-interface content comprises any of user controls, programmable buttons and action buttons;
- the controller determining which of the one or more actions is responsible for acting on the first user request;
- obtaining, using the one or more actions, one or more parameter values from the business objects and dynamically manipulating the one or more parameter values;
- using the one or more actions and the business object parameter values, selecting a first widget from the one or more widgets;

associating the first widget with the first panel selected from the one or more panels, wherein the first widget is arranged into a specified dynamic layout within the first panel; and
generating the specified dynamic layout, at runtime, and presenting to the user, a user-interface utility comprising the HTML user interface page that includes the first widget arranged into the specified dynamic layout within the first panel; wherein the first widget has the capability of representing properties of the business objects as HTML elements;
wherein the method performed by one or more processors.

3. (original) A method as recited in Claim 2, wherein the business object parameters are associated with one of the widgets based on the user request.
4. (original) A method as recited in Claim 2, wherein the application program is a network management application program.
5. (original) A method as recited in Claim 2, wherein receiving one or more business objects that define functions of the application program comprises receiving an XML file that defines the business objects and one or more of the parameters for the business objects.
6. (original) A method as recited in Claim 2, further comprising the step of generating, using the widget, client-side executable program code that performs one or more data validation or access control operations on user input for the user operation.
7. (Previously Presented) A method as recited in Claim 2, wherein the actions interact with the business objects through service object module interfaces that provide parameter values for the business objects to the actions.
8. (original) A method as recited in Claim 2, further comprising the steps of:
receiving user input in a field of the user interface that is associated with the widget,
wherein the user input is received in HTML elements of an HTTP request from a browser;

converting the user input from the HTML elements into one or more programmatic objects having an appropriate data type for use by the application program.

9. (original) A method as recited in Claim 2, further comprising the step of associating a first widget with a second widget, wherein the first widget and second widget are related by a containment hierarchy.

10. (original) A method as recited in Claim 2, wherein each of the widgets represents one or more properties of the business objects by an HTML element.

11. (original) A method as recited in Claim 2, wherein the step of generating an HTML user interface page that includes the panel further comprises generating an HTML user interface page that includes one or more of JSP files, static HTML elements, style sheets, or images.

12. (Canceled)

13. (Currently Amended) A computer-readable volatile or non-volatile storage medium storing one or more sequences of instructions ~~for generating a consistent user interface for an application program~~, which instructions, when executed by one or more processors, cause the one or more processors to ~~carry out~~ perform the steps of:

automatically generating a consistent user interface for an application program by
assisting a user with building an HTML user interface page by:

receiving one or more business objects that each define a user action for the application program;

receiving one or more metadata elements for dynamic content generation and defining parameters for the user actions of the one or more business object;

invoking a controller that is communicatively coupled to one or more actions, one or more widgets, and one or more panels;

receiving a first user request from the user through a browser used to interact with the application program and dispatching the user request to one or more of the actions;

wherein the first panel is dynamically generated using the one or more widgets

and comprises a programmed template with a user-interface content;
wherein the user-interface content comprises any of user controls, programmable
buttons and action buttons;

based on a second user request received from the user through the browser,
selecting a first panel from the one or more panels and including the first
panel in the HTML user interface page;

the controller determining which of the one or more actions is responsible for
acting on the first user request;

obtaining, using the one or more actions, one or more parameter values from the
business objects and dynamically manipulating the one or more parameter
values;

using the one or more actions and the business object parameter values, selecting
a first widget from the one or more widgets;

associating the first widget with the first panel selected from the one or more
panels, wherein the first widget is arranged into a specified dynamic
layout within the first panel; and

generating the specified dynamic layout, at runtime, and presenting to the user, a user-
interface utility comprising the HTML user interface page that includes the first
widget arranged into the specified dynamic layout within the first panel; wherein
the first widget has the capability of representing properties of the business
objects as HTML elements.

14. (previously presented) A computer-readable storage medium as recited in Claim 13,
wherein the business object parameters are associated with one of the widgets based on the user
request.

15. (previously presented) A computer-readable storage medium as recited in Claim 13,
wherein the application program is a network management application program.

16. (previously presented) A computer-readable storage medium as recited in Claim 13,
wherein receiving one or more business objects that define functions of the application program
comprises receiving an XML file that defines the business objects and one or more of the

parameters for the business objects.

17. (previously presented) A computer-readable storage medium as recited in Claim 13, further comprising instructions for performing the step of generating, using the widget, client-side executable program code that performs one or more data validation or access control operations on user input for the user operation.

18. (Previously Presented) A computer-readable storage medium as recited in Claim 13, wherein the actions interact with the business objects through service object module interfaces that provide parameter values for the business objects to the actions.

19. (previously presented) A computer-readable storage medium as recited in Claim 13, further comprising instructions for performing the steps of:
receiving user input in a field of the user interface that is associated with the widget,
wherein the user input is received in HTML elements of an HTTP request from a browser;
converting the user input from the HTML elements into one or more programmatic objects having an appropriate data type for use by the application program.

20. (previously presented) A computer-readable storage medium as recited in Claim 13, further comprising instructions for performing the step of associating a first widget with a second widget, wherein the first widget and second widget are related by a containment hierarchy.

21. (previously presented) A computer-readable storage medium as recited in Claim 13, wherein each of the widgets represents one or more properties of the business objects by an HTML element.

22. (previously presented) A computer-readable storage medium as recited in Claim 13, wherein the step of generating an HTML user interface page that includes the panel further comprises generating an HTML user interface page that includes one or more of JSP files, static HTML elements, style sheets, or images.

23. (Currently Amended) An apparatus ~~for generating a consistent user interface for an application program~~, comprising:

means for automatically generating a consistent user interface for an application program

by assisting a user with building an HTML user interface page, the means

including:

means for receiving one or more business objects that each define a user action
for the application program;

means for receiving one or more metadata elements for dynamic content
generation and defining parameters for the user actions of the one or more
business object;

means for invoking a controller that is communicatively coupled to one or more
actions, one or more widgets, and one or more panels;

means for receiving a first user request from the user through a browser used to
interact with the application program and dispatching the user request to
one or more of the actions, wherein the controller determines which of the
one or more actions is responsible for acting on the user request;

means for selecting, based on a second user request received from the user
through the browser, a first panel from the one or more panels and
including the first panel in the HTML user interface page;

wherein the first panel is dynamically generated using the one or more widgets
and comprises a programmed template with a user-interface content;

wherein the user-interface content comprises any of user controls, programmable
buttons and action buttons;

means for obtaining, using the one or more actions, one or more parameter values
from the business objects and dynamically manipulating the one or more
parameter values;

means for selecting a first widget from the one or more widgets, using the one or
more actions and the business object parameter values;

means for associating the first widget with the first panel selected from the one or
more panels, wherein the first widget is arranged into a specified dynamic
layout within the first panel; and

means for generating the specified dynamic layout, at runtime, and presenting to the user,

a user-interface utility comprising the HTML user interface page that includes the first widget arranged into the specified dynamic layout with in the first panel; wherein the first widget has the capability of representing properties of the business objects as HTML elements.

24. (original) An apparatus as recited in Claim 23, wherein the business object parameters are associated with one of the widgets based on the user request.
25. (original) An apparatus as recited in Claim 23, wherein the application program is a network management application program.
26. (original) An apparatus as recited in Claim 23, wherein the means for receiving one or more business objects that define functions of the application program comprises means for receiving an XML file that defines the business objects and one or more of the parameters for the business objects.
27. (original) An apparatus as recited in Claim 23, further comprising means for generating, using the widget, client-side executable program code that performs one or more data validation or access control operations on user input for the user operation.
28. (Previously Presented) An apparatus as recited in Claim 23, wherein the actions interact with the business objects through service object module interfaces that provide parameter values for the business objects to the actions.
29. (original) An apparatus as recited in Claim 23, further comprising:
 - means for receiving user input in a field of the user interface that is associated with the widget, wherein the user input is received in HTML elements of an HTTP request from a browser;
 - means for converting the user input from the HTML elements into one or more programmatic objects having an appropriate data type for use by the application program.

30. (original) An apparatus as recited in Claim 23, further comprising means for associating a first widget with a second widget, wherein the first widget and second widget are related by a containment hierarchy.
31. (original) An apparatus as recited in Claim 23, wherein each of the widgets represents one or more properties of the business objects by an HTML element.
32. (original) An apparatus as recited in Claim 23, wherein the means for generating an HTML user interface page that includes the panel further comprises means for generating an HTML user interface page that includes one or more of JSP files, static HTML elements, style sheets, or images.
33. (Currently Amended) An apparatus ~~for generating a consistent user interface for an application program~~, comprising:
 - a network interface that is coupled to the data network for receiving one or more packet flows therefrom;
 - a processor;
 - one or more stored sequences of instructions which, when executed by the processor, cause the processor to ~~carry out~~ perform the steps of:
 - automatically generating a consistent user interface for an application program by
 - assisting a user with building an HTML user interface page by:
 - receiving one or more business objects that each define a user action for the application program;
 - receiving one or more metadata elements for dynamic content generation and defining parameters for the user actions of the one or more business object;
 - invoking a controller that is communicatively coupled to one or more actions, one or more widgets, and one or more panels;
 - receiving a first user request from the user through a browser used to interact with the application program and dispatching the user request to one or more of the actions;
 - based on a second user request received from the user through the browser,

selecting a first panel from the one or more panels and including the first panel in the HTML user interface page;
wherein the first panel is dynamically generated using the one or more widgets and comprises a programmed template with a user-interface content;
wherein the user-interface content comprises any of user controls, programmable buttons and action buttons;
the controller determining which of the one or more actions is responsible for acting on the first user request;
obtaining, using the one or more actions, one or more parameter values from the business objects and dynamically manipulating the one or more parameter values;
using the one or more actions and the business object parameter values, selecting a first widget from the one or more widgets;
associating the first widget with the first panel selected from the one or more panels, wherein the first widget is arranged into a specified dynamic layout within the first panel; and
generating the specified dynamic layout, at runtime, and presenting to the user, a user-interface utility comprising the HTML user interface page that includes the first widget arranged into the specified dynamic layout within the first panel; wherein the first widget has the capability of representing properties of the business objects as HTML elements.

34. (original) An apparatus as recited in Claim 33, wherein the business object parameters are associated with one of the widgets based on the user request.

35. (original) An apparatus as recited in Claim 33, wherein the application program is a network management application program.

36. (original) An apparatus as recited in Claim 33, wherein receiving one or more business objects that define functions of the application program comprises receiving an XML file that defines the business objects and one or more of the parameters for the business objects.

37. (original) An apparatus as recited in Claim 33, further comprising instructions for performing the step of generating, using the widget, client-side executable program code that performs one or more data validation or access control operations on user input for the user operation.

38. (Previously Presented) An apparatus as recited in Claim 33, wherein the actions interact with the business objects through service object module interfaces that provide parameter values for the business objects to the actions.

39. (original) An apparatus as recited in Claim 33, further comprising instructions for performing the steps of:
receiving user input in a field of the user interface that is associated with the widget,
wherein the user input is received in HTML elements of an HTTP request from a browser;
converting the user input from the HTML elements into one or more programmatic objects having an appropriate data type for use by the application program.

40. (original) An apparatus as recited in Claim 33, further comprising instructions for performing the step of associating a first widget with a second widget, wherein the first widget and second widget are related by a containment hierarchy.

41. (original) An apparatus as recited in Claim 33, wherein each of the widgets represents one or more properties of the business objects by an HTML element.

42. (original) An apparatus as recited in Claim 33, wherein the instructions for performing the step of generating an HTML user interface page that includes the panel further comprises generating an HTML user interface page that includes one or more of JSP files, static HTML elements, style sheets, or images.

43. – 45. (Canceled)